

91244
May 26, 1992
P-15

Final report for NASA Contract NAG-1-1079.

Funding period: 1/16/90-5/31/92.

OPTIMIZATION BY NONHIERARCHICAL ASYNCHRONOUS DECOMPOSITION

Jayashree Shankar

Department of Computer Science

Calvin J. Ribbens

Department of Computer Science

Raphael T. Haftka

Department of Aerospace and Ocean Engineering

Layne T. Watson

Interdisciplinary Center for Applied Mathematics

Virginia Polytechnic Institute and State University

Blacksburg, VA 24061-0106

Abstract. Large scale optimization problems are tractable only if they are somehow decomposed. Hierarchical decompositions are inappropriate for some types of problems and do not parallelize well. Sobieszczanski-Sobieski has proposed a nonhierarchical decomposition strategy for nonlinear constrained optimization that is naturally parallel. Despite some successes on engineering problems, the algorithm as originally proposed fails on simple two dimensional quadratic programs. This paper carefully analyzes the algorithm for quadratic programs, and suggests a number of modifications to improve its robustness.

1. Introduction.

Many engineering problems involve large scale optimization over many different disciplines. As is the case with many large scale problems, a decomposition of the problem into subproblems helps reduce the time and complexity of solution. The strategy governing the decomposition of a large scale problem can directly affect the ease and accuracy of the solution. The concept of a linear decomposition strategy [23] has been used with good results in a number of cases. This method works very well in the case of a system that is amenable to such a decomposition, i.e., when subsystems can be laid out clearly in a linear fashion. For this there should be a basic hierarchy embedded in the problem.

For a system with many interdependencies between the probable subproblems, using a linear decomposition strategy implies choosing one subsystem before another, thereby establishing an artificial hierarchy. The order chosen will affect the solution iterates, making this strategy ill-suited or even nonconvergent for such nonhierarchic problems.

These considerations led Sobieski [22] to propose a new nonhierarchic decomposition strategy. Since nonlinear optimization can be reduced to a series of quadratic programs, it is appropriate to study this new algorithm first on quadratic programs. Thus, this paper first studies the various tuning parameters occurring in this algorithm, using a model quadratic programming problem. A series of experiments shows that modifications to the algorithm as originally proposed by Sobieski [22] are necessary for convergence. This modified algorithm is then used to solve problems involving a number of subsystems, each with a varying number of design variables.

The tests are carried out on quadratic programming (QP) problems of different dimensions. The decomposition then yields subproblems which are also QP problems. The method employed to solve these smaller QP problems is elimination of variables [8]. Also optimization packages such as MINOS [17] and QPSOL [12] were used to verify the correct answers.

A detailed description of the original algorithm, modifications to it, tabulations of the results obtained for the test problems of different dimensions, and analysis of the results are presented.

2. Problem Statement.

Consider the following nonlinear programming problem (NLP),

$$\begin{aligned} & \min_x \Theta(x) \\ & \text{subject to } g(x, y) \leq 0, \\ & h(x, y) = 0, \end{aligned}$$

where $x \in E^n$, $y \in E^p$, g is an m -dimensional vector function and h is a p -dimensional vector function. x is the set of design variables and y is the set of behavior variables which are the unknowns in each subsystem.

The approach (known as subspace optimization) is to solve this problem by solving a set of subproblems. To outline the differences between the current scheme and simple decomposition, we introduce the following terminology:

$$\begin{aligned} x &= (X^1, X^2, \dots, X^N), & X^i &\in E^{n_i}, & n_1 + n_2 + \dots + n_N &= n, \\ y &= (Y^1, Y^2, \dots, Y^N), & Y^i &\in E^{p_i}, & p_1 + p_2 + \dots + p_N &= p, \\ g &= \begin{pmatrix} g^1 \\ \vdots \\ g^N \end{pmatrix}, & h &= \begin{pmatrix} h^1 \\ \vdots \\ h^N \end{pmatrix}, \\ h^i(x, y) &\in E^{p_i}, & g^i(x, y) &\in E^{m_i}, & m_1 + \dots + m_N &= m, \\ h^i(x, y) &= Y^i - \tilde{h}^i(x, Y^1, \dots, Y^{i-1}, Y^{i+1}, \dots, Y^N). \end{aligned}$$

The sub vector X^i is the set of design variables corresponding to the i th subsystem. Similarly the sub vector Y^i is the set of behavior variables of the i th subsystem. For any vector function $f(x, y)$, let $\hat{f}(X^i, Y^i)$ denote f with all the components $X^1, \dots, X^{i-1}, X^{i+1}, \dots, X^N, Y^1, \dots, Y^{i-1}, Y^{i+1}, \dots, Y^N$ fixed except for X^i and Y^i . Note the assumption that each Y^i can be explicitly determined in terms of x and the other subvectors Y^j .

3. Simple Sequential Decomposition.

The approach is to first divide the given large problem into a set of independent subproblems, corresponding naturally to the subsystems comprising the larger system. The i th subsystem would be

$$\begin{aligned} & \min_{X^i} \hat{\Theta}(X^i) \\ & \text{subject to } \hat{g}^i(X^i, Y^i) \leq 0, \\ & \hat{h}^i(X^i, Y^i) = 0, \end{aligned}$$

where the system of equalities $\hat{h}^i = 0$ is used to eliminate Y^i from \hat{g}^i . The subproblems are solved sequentially for $i = 1, \dots, N$, with one pass through all the subsystems constituting one outer iteration. The outer iterations are repeated until the same point (\bar{x}, \bar{y}) solves all N subproblems. While solving the i th subsystem the values of $X^1, \dots, X^{i-1}, X^{i+1}, \dots, X^N, Y^1, \dots, Y^{i-1}, Y^{i+1}, \dots, Y^N$ are fixed. They can be chosen in a Gauss-Seidel manner where the first $i - 1$ X and Y subvectors used have their latest values from solving the first $i - 1$ subproblems. A parallel algorithm, solving the subproblems concurrently, would use a Jacobi scheme where the values of all the X^j and Y^j vectors are updated only at the end of each major outer iteration. The ensuing discussion assumes a Jacobi scheme.

4. Decomposition with Approximate Coupling.

In the scheme proposed by Sobieski [22], a measure of the constraints in each of the other subsystems is also brought into the i th subsystem in the form of one cumulative constraint C_i^k per subsystem. The approximate cumulative constraint C_i^k of the k th subsystem in the i th subsystem is obtained from the corresponding constraints $g^k \in E^{m_k}$ as a linearization of the Kreisselmeier-Steinhauser cumulative constraint

$$K_k(x, y) = \frac{1}{\rho} \ln \left(\sum_{j=1}^{m_k} e^{\rho g_j^k(x, y)} \right).$$

The ρ in the Kreisselmeier-Steinhauser function is a constant used to control the accuracy of the cumulative constraint approximation. The linearization of this cumulative constraint of the k th subsystem with respect to the variables of the i th subsystem is

$$C_i^k(X^i, Y^i) = \hat{K}_k(X_0^i, Y_0^i) + \sum_{j=1}^{n_i} \frac{\partial \hat{K}_k}{\partial X_j^i}(X_0^i, Y_0^i) (X_j^i - (X_0^i)_j).$$

In the i th subsystem the cumulative constraints of the other subsystems are brought in as constraints. Therefore, a violated cumulative constraint of one subsystem may be satisfied by decisions taken in every one of the other subsystems. Therefore, we introduce coefficients r_i^p to represent the fractional "responsibility" assigned to the i th subsystem for reducing the violation of the cumulative constraint of the p th subsystem, for each $p = 1, \dots, N$. Thus we have N^2 r -coefficients. The r_i^p 's are defined in such a way that

$$\sum_{i=1}^N r_i^p = 1,$$

Sobieski [22] suggested the initialization of the r -coefficients in such a way that they are proportional to the degree of influence exerted by the i th subsystem on the p th cumulative constraint. This initialization is discussed in the Appendix.

To further reduce the objective function we allow cumulative constraints to be violated in one subsystem, provided that the violation will be offset by oversatisfaction of that constraint in another subsystem. To account for such tradeoffs, we introduce the N^2 coefficients t_i^p , corresponding to

the cumulative constraint of the p th subsystem when present in the i th subsystem. For the p th cumulative constraint,

$$\sum_{i=1}^N t_i^p = 0,$$

maintains the constraint at a value of zero. This condition and the condition on the r -coefficients are enforced in what is called the coordination optimization phase, which is solved to update the values of the r 's and the t 's at the end of every outer iteration. The t_i^p 's are initialized at the beginning of the algorithm to zero.

As has been described above, the r_i^p 's are needed only in the case of a violation and the t_i^p 's only when the constraints are critical, therefore only one of the two is needed at a time. Therefore we introduce N coefficients s^p which act as switches, one for each of the cumulative constraints of the subsystems. s^p is set to one (activating the r -coefficients) if the corresponding constraint $K_p \leq 0$ is violated at the outset of the system optimization procedure and stays at one until the K_p is driven to a critical status (zero value). Once K_p becomes critical, s^p is reset to zero (activating the t -coefficients) and stays at zero until the system optimization procedure terminates. The switch s^i is applied selectively to the natural constraints g^i of the i th subsystem (i.e., the constraints that are assigned to the i th subsystem) by multiplying the r -coefficient r_i^i by a factor of $\max\{\hat{g}^i(X_0^i, Y_0^i), 0\}$, so that constraints which are already satisfied are not taken into consideration.

Thus, the i th subsystem optimization problem is

$$\begin{aligned} & \min_{X^i} \hat{\Theta}(X^i) \\ \text{subject to } & \hat{g}^i(X^i, Y^i) \leq s^i \max\{\hat{g}^i(X_0^i, Y_0^i), 0\}(1 - r_i^i) + (1 - s^i)t_i^i, \\ & C_i^p(X^i, Y^i) \leq \hat{K}_p(X_0^i, Y_0^i) s^p(1 - r_i^p) + (1 - s^p)t_i^p, \\ & \qquad \qquad \qquad p = 1, \dots, i-1, i+1, \dots, N, \\ & \hat{h}^i(X^i, Y^i) = 0. \end{aligned}$$

The constrained minimum of Θ obtained from each subsystem optimization is a function of the constants r_i^p and t_i^p , and its partial derivatives with respect to r_i^p and t_i^p (assuming they exist) can be computed from the expressions given in the Appendix using gradient information for the Θ and C functions. These derivatives are used for a linear approximation of Θ that is the objective function for the *coordination optimization phase*, the last (and synchronizing) step of an outer iteration.

The coordination optimization phase (COP) solves a linear program to adjust the coefficients r_i^p and t_i^p , so that the objective function Θ will be further reduced (if possible) at the end of the next outer iteration. The linear program uses a linear extrapolation of Θ based on the partial derivatives $\partial\Theta/\partial z$ described above. Here z represents either an r or a t coefficient. Move limits (upper and lower bounds U_i^p , \bar{U}_i^p , L_i^p and \bar{L}_i^p for r_i^p and t_i^p , respectively) are needed to prevent large changes in the r - and t -coefficients caused by the nonlinearity of the original problem. For the first COP execution, the r_i^p 's may be initialized as already suggested and the t_i^p 's are initialized to zero. For every subsequent execution, the r_i^p 's and the t_i^p 's are initialized to the terminal values from the previous COP execution. The result of the COP execution is a new set of r_i^p 's and t_i^p 's to be used in the next outer loop of subsystem optimizations. The adjustment of the r_i^p 's

and t_i^p 's to the new values amounts to a reassignment of the responsibility for eliminating the constraint violations among the subsystems and to issuing a new set of instructions about trading the constraint violations/oversatisfactions among these subsystems. Let (x_0, y_0) be the current updated point (the result of the Jacobi outer iteration) and

$$\Theta_1 = \Theta(x_0, y_0) + \sum_{p=1}^N \sum_{i=1}^N \frac{\partial \Theta}{\partial r_i^p} \Delta r_i^p + \sum_{p=1}^N \sum_{i=1}^N \frac{\partial \Theta}{\partial t_i^p} \Delta t_i^p,$$

where $\Delta r_i^p = (r_i^p - (r_i^p)_0)$ and $\Delta t_i^p = (t_i^p - (t_i^p)_0)$. The partial derivatives $\partial \Theta / \partial r_i^p$ and $\partial \Theta / \partial t_i^p$ are evaluated at the optimal point computed by the i th subsystem optimization. Let

$$R = (r_1^1, r_1^2, \dots, r_1^N, r_2^1, \dots, r_2^N, \dots, r_N^1, \dots, r_N^N)$$

and

$$T = (t_1^1, t_1^2, \dots, t_1^N, t_2^1, \dots, t_2^N, \dots, t_N^1, \dots, t_N^N).$$

Then Θ_1 is a function of R and T . The linear program solved during the coordination optimization phase is:

$$\begin{aligned} & \min_{R, T} \Theta_1(R, T) \\ & \text{subject to} \quad \sum_{k=1}^N r_k^p = 1, \quad p = 1, \dots, N, \\ & \quad \quad \quad \sum_{k=1}^N t_k^p = 0, \quad p = 1, \dots, N, \\ & \quad \quad \quad 0 \leq r_k^p \leq 1, \quad p = 1, \dots, N, \quad k = 1, \dots, N, \\ & \quad \quad \quad L_k^p \leq r_k^p \leq U_k^p, \quad p = 1, \dots, N, \quad k = 1, \dots, N, \\ & \quad \quad \quad \tilde{L}_k^p \leq t_k^p \leq \tilde{U}_k^p, \quad p = 1, \dots, N, \quad k = 1, \dots, N. \end{aligned}$$

5. Pseudocode for algorithm.

An algorithmic description of the whole process in pseudo-code is given next, using the following model quadratic programming problem (without the variables y and equality constraints $h(x, y) = 0$) for specificity:

$$\begin{aligned} & \min_x x^t A x \\ & \text{subject to} \quad Bx \leq d, \end{aligned}$$

where

$$A = \begin{pmatrix} A_{11} & \alpha_{12} A_{12} & \dots & \alpha_{1N} A_{1N} \\ \alpha_{12} A_{12}^t & A_{22} & \dots & \alpha_{2N} A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{1N} A_{1N}^t & \alpha_{2N} A_{2N}^t & \dots & A_{NN} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & \beta_{12} B_{12} & \dots & \beta_{1N} B_{1N} \\ \beta_{21} B_{21} & B_{22} & \dots & \beta_{2N} B_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{N1} B_{N1} & \beta_{N2} B_{N2} & \dots & B_{NN} \end{pmatrix},$$

$$d = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{pmatrix}, \quad x = \begin{pmatrix} X^1 \\ X^2 \\ \vdots \\ X^N \end{pmatrix},$$

and $X^i \in E^{n_i}$, $A_{ij} \in E^{n_i \times n_j}$, $B_{ij} \in E^{p_i \times n_j}$, $d_i \in E^{p_i}$, the A_{ii} are symmetric and positive definite and the α_{ij} , β_{ij} are fixed “coupling” parameters, for all $i, j = 1, \dots, N$. Using notation defined in the Appendix, pseudo-code for the algorithm applied to this quadratic programming problem (QP) is:

Choose an initial estimate x and initialize the r -, s - and t -coefficients;

Repeat until minimum reached

begin

for $i = 1$ to N **do**

begin

Calculate the linearization $C_i^j(X^i)$ of the cumulative constraint for the j th subsystem, for all $j \neq i$;

Calculate the i th subsystem’s self responsibility

$$\delta^i = s^i \max \{ \hat{g}^i(X_0^i, Y_0^i), 0 \} (1 - r_i^i) + (1 - s^i) t_i^i;$$

Solve the QP (i th subsystem)

$$\min_{X^i} \hat{\Theta}(X^i) = (X^i)^t A_{ii} X^i + 2 \left(\sum_{j \neq i} \alpha_{ij} (X^i)^t A_{ij} X^j \right)$$

$$\text{subject to } \sum_{j=1}^N \beta_{ij} B_{ij} X^j - d_i \leq \delta^i, \quad (\beta_{ii} = 1)$$

$$\tilde{C}_i^j(X^i) \leq 0, \quad \text{for all } j \neq i;$$

Calculate (if not already available) the Lagrange multipliers λ using the method given in the Appendix.

Calculate $\frac{\partial \Theta}{\partial r_i^j}$ and $\frac{\partial \Theta}{\partial t_i^j}$ for $j = 1, \dots, N$;

end

Solve the LP (Coordination Optimization Phase)

$$\min_{R, T} \Theta_1(R, T)$$

$$\begin{aligned} \text{subject to } \quad & \sum_{k=1}^N r_k^p = 1, \quad \sum_{k=1}^N t_k^p = 0, \quad p = 1, \dots, N, \\ & 0 \leq r_k^p \leq 1, \quad L_k^p \leq r_k^p \leq U_k^p, \quad \tilde{L}_k^p \leq t_k^p \leq \tilde{U}_k^p, \\ & p = 1, \dots, N, \quad k = 1, \dots, N; \end{aligned}$$

end (repeat)

6. Initial tests.

Testing of this algorithm was first performed on a simple 2×2 case:

Example 1.

$$\begin{aligned} \min_x \quad & x_1^2 + x_2^2 \\ \text{subject to} \quad & x_1 + \beta x_2 \leq 4, \\ & \beta x_1 + x_2 \geq 2, \\ \text{where } x = (x_1, x_2)^t \in E^2. \end{aligned}$$

Here each constraint is taken to be in a subsystem by itself with $X^1 = (x_1)$ and $X^2 = (x_2)$. The results are tabulated in Table I. The column headings are the starting points, the last column gives the solutions for the different values of β , and each entry contains a convergence code and the number of iterations taken. The code IF means an infeasible subproblem is encountered at the very first iteration and the procedure is terminated, R means the solution is reached in the iteration indicated, but subsequently an infeasible subproblem is encountered, WR means a wrong point is reached before an infeasible subproblem causes termination, O means there is oscillation through the number of iterations indicated, WC means there is convergence to a point other than the solution, and NC means there is no convergence even after the number of iterations indicated.

TABLE I
Original algorithm applied to the 2×2 case.

β	(2,3)	(4,-1)	(1,-1)	(0.8,1.5)	(10,3)	solution
0.0	R 1	R 1	R 1	R 1	R 1	(0.0,2.0)
0.1	WR 1	IF 1	WC 4	WC 4	IF 1	(0.198,1.98)
0.3	WR 1	IF 1	WC 5	WC 4	IF 1	(0.55,1.835)
0.5	WR 1	IF 1	O 150	NC 150	WR 1	(0.8,1.6)
1.0	O 150	O 150	O 150	O 150	O 150	(1.0,1.0)

As can be seen from the table, the main problem with the algorithm is not being able to deal with infeasibility in a subproblem. Another reason for the algorithm not being successful is the way the s coefficient is set permanently to zero once a constraint becomes critical. When a constraint that was once critical becomes violated, the r coefficients cannot be brought in to reduce the violation.

7. Modifications to the original algorithm.

Several modifications and variations of the original algorithm as described in [22] are discussed next. The order of the topics is not significant.

Changes in setting of the switch coefficients s^p .

When a K_p becomes critical, the corresponding s^p is set to zero and stays at zero until the whole procedure terminates. This means that the term with the r^p coefficient does not contribute any longer to the constraints. If the constraint becomes violated later, then the violation cannot be reduced using the r coefficients. Hence two alternatives to the algorithm were considered. One was to remove the s coefficient from the r term. The other was to set the s^p coefficient at the end of every outer iteration depending on whether the corresponding constraint was satisfied or not. This would make one of the r or t coefficients active all the time. The second alternative performed better in initial tests and therefore was selected.

Handling Infeasibility.

Because of the linearization and the allocation of responsibility, some subproblems may be infeasible. The following procedure is employed to recover from infeasibility in a subsystem. A new variable ω is introduced in each of the constraints and a large multiple of this variable is added to the objective function to be minimized. Thus, the corresponding sub problem would now be

$$\begin{aligned} & \min_{X^i} \hat{\Theta}(X^i) + M\omega \\ \text{subject to} \quad & \hat{g}^i(X^i, Y^i) - \omega \leq s^i \max\{\hat{g}^i(X_0^i, Y_0^i), 0\}(1 - r_i^i) + (1 - s^i)t_i^i, \\ & C_i^p(X^i, Y^i) - \omega \leq \hat{K}_p(X_0^i, Y_0^i) s^p(1 - r_i^p) + (1 - s^p)t_i^p, \\ & p = 1, \dots, i-1, i+1, \dots, N, \end{aligned}$$

where M is a large positive number.

The linear objective function in the Coordination Optimization Phase is formed using the Lagrange multipliers obtained from the subproblems. The introduction of the variable ω affects these Lagrange multipliers. The sensitivity derivatives are affected considerably because of these Lagrange multipliers, as indicated in the Appendix. A thought as to whether this was justified or not led to two variations of the algorithm. In the first variation, in case of an infeasibility in any subproblem, the Coordination Optimization Phase is omitted at the end of that outer iteration. In the second variation, the COP is included in every outer iteration.

Limit on t coefficients.

Initially tests were performed with the t coefficients left unbounded, but clearly this is unwise. A few variations for the bounds on the t 's were considered. One possibility is keeping the bound fixed throughout the procedure, but this may result in nonconvergence to the solution. Also the bounds should not decrease too fast, because this may force convergence to a nonoptimal solution.

The bound is reduced by a factor of $f = 0.8$ at the end of every outer iteration. Thus, if the bound at the first iteration is t_1 , then the bound at the m th iteration is

$$t_m = 0.8^{(m-1)} t_1.$$

A variation of having the bound at the m th iteration equal a factor $f^{\log(m-1)}$ or a factor $f^{(m-1)^{1/2}}$ of the bound at the first iteration was also considered.

A later modification was to change the move limits on the t coefficients based on information about the corresponding coefficients in the objective function of the COP. To ensure that no subsystem is allowed a violation that cannot be offset by an equivalent oversatisfaction in the other subsystems, a change was made to the move limits on the t coefficients, using information about the corresponding sensitivities.

The a^{pk} coefficient (as described in the Appendix) is a measure of the sensitivity of the p th cumulative constraint to the variables of the k th subsystem. The lower limit of t_k^p is now $\max\{-a^{pk}, -tlim\}$, reasoning that the oversatisfaction expected of the p th cumulative constraint in the k th subsystem will be restricted to what it can handle.

The initial value of the move limit on the t coefficient affects the results and the path taken. Various values of this initial limit were tried.

At the beginning of every major iteration involving a new ρ , the initial value of the move limit on the t coefficient may be reset to either the original value or some fraction of it.

Convergence Criterion.

The convergence criterion initially involved a measure of the difference between three successive iteration values of the design vector. Later, tests revealed that with this criterion the procedure could stop even if there was a chance for further improvement, because of changes in the values of the r and t coefficients. Hence the difference between the values of the t and r coefficients were also included in the convergence criterion. Thus, if S_{m-2} represents the normalized form of the vector (x, R, T) at the $(m-2)$ th iteration and similarly for S_{m-1} and S_m , then using the 2-norm the convergence criterion is

$$\|S_{m-1} - S_{m-2}\| + \|S_m - S_{m-1}\| \leq 0.0001.$$

Changes to the ρ coefficient.

After the required convergence criterion is met the ρ coefficient is increased and the whole process is repeated again to check if the convergence criterion is still met, if not another major iteration is performed. This is because with increasing ρ the cumulative constraint is closer to the actual constraints. But the process is not started with a large ρ , as the problem is then very ill conditioned.

Cross derivatives.

The cross derivatives are checked to see if one subsystem is at all dependent on the variables of another subsystem, if not the corresponding r and t coefficients are fixed at zero.

Changes to the r coefficients.

The diagonal r coefficients r_i^i are assigned a minimum value of 0.2 always, reasoning that every subsystem always has some responsibility towards its own constraints.

No COP.

If the objective function of the COP is a constant then the COP is skipped for that iteration.

Resetting the t coefficients.

It was observed in one case that t coefficients (corresponding to one subsystem's constraint) with equal derivatives in the objective function of the COP assigned extreme values to the corresponding t coefficients even though there was no contribution to the objective function. Hence, after the COP a check is performed on the t coefficients to see if for a particular p the sum of the contributions of all the corresponding coefficients to the objective function of the COP is zero. If so all the t coefficients corresponding to this p are forced to be zero. This check is performed for all values of p .

Different combinations of these modifications were used on the following test problems and the results are given in the tables following them.

8. Further tests.

The original algorithm as proposed by Sobieszczanski-Sobieski [22] did not prove to be successful as indicated by Table I. Results of two of the most successful variations to this algorithm tested on the 2×2 case are tabulated in Tables II and III. The characteristics of the algorithms used are given above the corresponding tables. "s updated" indicates that the s coefficient is updated at the end of every outer iteration as indicated in the modifications given in Section 7. " ω used" means that an artificial variable ω was introduced to deal with infeasible subproblems.

The inclusion or exclusion of the COP is in the case of an infeasibility in any subproblem. The limit on the magnitude of the t coefficient is bounded at 1 initially and this bound is decreased by using a factor of 0.8 as described in Section 7. The most successful version was used for larger test problems like the 3×3 case with two subsystems and the 6×6 case with three subsystems. The tests were carried out for five different values of β and for five different starting points. The column headings are the starting points, the last column gives the solutions for the different values of β , and each entry contains a convergence code and the number of iterations until the two-norm of the change in (x, R, T) is less than 0.0001. The code C means there is convergence to the solution, WC means there is convergence but not to the solution, and NC means there is no convergence even in the specified number of iterations.

TABLE II
*s updated, ω used, no COP,
 t bounded at 1 and 0.8 update.*

β	(2,3)	(4,-1)	(1,-1)	(0.8,1.5)	(10,3)	solution
0.0	C 6	C 6	C 6	C 6	C 6	(0.0,2.0)
0.1	C 56	C 54	C 53	C 53	C 55	(0.198,1.98)
0.3	C 50	C 57	C 51	C 52	C 44	(0.55,1.835)
0.5	C 7	C 55	C 42	C 48	C 49	(0.8,1.6)
1.0	C 7	C 8	C 7	C 47	C 46	(1.0,1.0)

TABLE III
*s updated, ω used, COP,
 t bounded at 1 and 0.8 update.*

β	(2,3)	(4,-1)	(1,-1)	(0.8,1.5)	(10,3)	solution
0.0	C 6	C 6	C 6	C 6	C 6	(0.0,2.0)
0.1	C 56	C 53	C 53	C 53	C 56	(0.198,1.98)
0.3	C 50	C 50	C 51	C 52	C 50	(0.55,1.835)
0.5	C 7	C 45	C 42	C 48	C 49	(0.8,1.6)
1.0	C 7	C 8	C 7	C 47	C 46	(1.0,1.0)

Example 2.

$$\begin{aligned}
& \min_x \quad x_1^2 + x_2^2 + x_3^2 \\
& \text{subject to} \quad x_1 + x_2 + \beta x_3 \leq 4, \\
& \quad \quad \quad -x_1 - x_2 - \beta x_3 \leq -2, \\
& \quad \quad \quad -\beta x_1 - \beta x_2 - 5x_3 \leq -2, \\
& \text{where } x = (x_1, x_2, x_3)^t \in E^3
\end{aligned}$$

TABLE IV
s updated, ω used, COP,
t bounded at 1 and 0.8 update.

β	(0,1,-3)	(1,1,0)	(4,0.1,0.8)	(-10,3,-10)	(0,0,0)	solution
0.0	C 6	C 6	C 6	C 6	C 6	(1,1,0.4)
0.1	C 60	C 60	C 59	C 60	C 59	(0.9819,0.9819,0.3607)
0.3	C 55	C 56	C 56	C 56	C 56	(0.9569,0.9569,0.2870)
0.5	WC 54	C 45	WC 54	C 47	WC 56	(0.8888,0.8888,0.4444)
1.0	C 47	C 46	C 46	WC 51	C 44	(0.6666,0.6666,0.6666)

Here, the first two constraints belong to one subsystem and the third constraint to another subsystem, $X^1 = (x_1, x_2)$ and $X^2 = (x_3)$.

It was observed that changing the initial value of the bound on t affects the convergence of the algorithm. For the WC entries in Table IV corresponding to $\beta = 0.5$, starting with a bound of 100 on t results in convergence to the solution for two of the three cases, and 1000 works for the third case. An initial bound of 10 for t leads to convergence to the solution for the WC case with $\beta = 1.0$. However, initial bounds of 10, 100, or 1000 cause failure for other cases in Table IV.

Example 3.

$$\begin{aligned}
& \min_x \quad x_1^2 + x_2^2 + x_3^2 + 2.5x_4^2 + 2.5x_5^2 + 10x_6^2 \\
& \text{subject to} \quad x_1 + x_2 + x_3 + 0 - \beta x_5 - 2\beta x_6 \leq 4, \\
& \quad \quad \quad -x_1 - x_2 - x_3 - \beta x_4 + 0 + 0 \leq -2, \\
& \quad \quad \quad -x_1 - x_2 - 5x_3 + 0 + 0 + 0 \leq -2, \\
& \quad \quad \quad 0 + 0 + 0 + x_4 + x_5 - \beta x_6 \leq -4, \\
& \quad \quad \quad \beta x_1 + \beta x_2 + 0 - 5x_4 - 4x_5 - \beta x_6 \leq 20, \\
& \quad \quad \quad \beta x_1 + \beta x_2 - \beta x_3 + 0 + 0 - x_6 \leq -6, \\
& \quad \quad \quad \text{where } x = (x_1, x_2, x_3, x_4, x_5, x_6)^t \in E^6
\end{aligned}$$

Here there are three subsystems with $n_1 = 3$, $n_2 = 2$ and $n_3 = 1$.

TABLE V
s updated ω used, COP,
t bounded at 1 and 0.8 update.

β	(0,0,0, 0,0,0)	(1,2,3, -1,1,5)	(-10,4,4, 0.8,0.1,1)	(1,1,1, 1,1,1)	(-4,2,2, 0,1,1)	solution
0.0	C 6	C 6	C 6	C 6	C 6	(0.6,0.6,0.6,-2.0,-2.0,6.0)
0.1	WC 46	WC 80	WC 48	NC 70	WC 48	(-2.4,-2.4,7.0,-1.7,-1.8,4.8)
0.3	WC 51	NC 70	WC 50	NC 70	WC 50	(-2.7,-2.7,8.0,-1.5,-1.8,1.9)
0.5	WC 52	NC 70	WC 53	NC 70	WC 53	(-1.7,-1.7,6.3,-1.5,-1.9,1.0)
0.5	WC 16	WC 7	WC 13	NC 70	WC 13	(-0.5,-0.5,4.2,-1.2,-2.0,0.7)

Appendix.

Initialization of the r -coefficients.

The coefficients may be initialized on the basis of sensitivity information so as to assign a greater responsibility for a cumulative constraint satisfaction (of the i th subsystem say) to those subsystems that have a greater influence on that constraint. Let

$$(K_{pk})_i \equiv \frac{\partial K_p}{\partial X_i^k}(x_0, y_0).$$

Since $1 \leq p \leq N$, $1 \leq k \leq N$, and $1 \leq i \leq n_k$, there are Nn_k such partial derivatives $(K_{pk})_i$, for every k . Now define

$$a^{pk} \equiv \max_{1 \leq i \leq n_k} |(K_{pk})_i|, \quad 1 \leq p \leq N, \quad 1 \leq k \leq N,$$

which measures the influence of the k th subsystem's variables X^k on the p th subsystem's constraints, as represented by K_p . Normalizing these N^2 influence coefficients gives the r -coefficients

$$r_k^p = \frac{a^{pk}}{\sum_{j=1}^N a^{pj}}, \quad 1 \leq p \leq N, \quad 1 \leq k \leq N.$$

Optimum Sensitivity Analysis.

Let z denote either of r_i^p or t_i^p , and define the modified constraint functions

$$\begin{aligned} \tilde{g}^i(X^i, Y^i) &= \hat{g}^i(X^i, Y^i) - [s^i \max\{\hat{g}^i(X_0^i, Y_0^i), 0\}(1 - r_i^i) + (1 - s^i)t_i^i], \\ \tilde{C}_i^p(X^i, Y^i) &= C_i^p(X^i, Y^i) - [\hat{K}_p(X_0^i, Y_0^i) s^p(1 - r_i^p) + (1 - s^p)t_i^p], \\ i &= 1, \dots, N, \quad p = 1, \dots, i-1, i+1, \dots, N. \end{aligned}$$

Let $\nabla_i = \left(\frac{\partial}{\partial X_1^i}, \dots, \frac{\partial}{\partial X_{n_i}^i} \right)$,

$$G^i = \begin{pmatrix} \tilde{g}_i^i \\ \tilde{C}_i^1 \\ \vdots \\ \tilde{C}_i^{i-1} \\ \tilde{C}_i^{i+1} \\ \vdots \\ \tilde{C}_i^N \end{pmatrix},$$

and G_A^i denote the subvector of G^i corresponding to the active constraints at the current point. It is assumed that the dimension of G_A^i is less than or equal to n_i , and that the Jacobian matrix $\nabla_i G_A^i$ has full rank. Then the sensitivities of the minimum of Θ with respect to the constraints $\tilde{g}_i^i \leq 0$, $\tilde{C}_i^p \leq 0$ are given by the Lagrange multipliers

$$\lambda = - \left[(\nabla_i G_A^i) (\nabla_i G_A^i)^t \right]^{-1} (\nabla_i G_A^i) (\nabla_i \Theta)^t,$$

where everything is evaluated at the current point—the result of the N th subsystem optimization. Now from this the sensitivities of the minimum of Θ with respect to the τ_i^p and t_i^p are given by

$$\frac{\partial \Theta}{\partial z} = \lambda^t \frac{\partial G_A^i}{\partial z}.$$

Observe that from the form of G^i , the partials $\partial G_A^i / \partial z$ are trivial to compute. λ would not be computed explicitly from the projection operator as described above, but rather from a QR factorization of $(\nabla_i G_A^i)^t$, as described in Fletcher [8].

REFERENCES

- [1] F. F. ABDI, H. IDE, V. J. SHANKAR, AND J. SOBIESZCZANSKI-SOBIESKI, *Optimization for nonlinear aeroelastic tailoring criteria*, Int. Coun. Aero. Sci., 16th Congress, Aug., 1988, Jerusalem.
- [2] J.-F. M. BARTHELEMY AND J. SOBIESZCZANSKI-SOBIESKI, *Optimum sensitivity derivatives of objective functions in nonlinear programming*, AIAA J., 22 (1983), pp. 913–915.
- [3] J.-F. M. BARTHELEMY AND M. F. RILEY, *Improved multi-level optimization approach for the design of complex engineering systems*, AIAA J., 26 (1988), pp. 353–360.
- [4] J.-F. M. BARTHELEMY, *Engineering applications of multilevel optimization*, Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Sept., 1988, Hampton, VA.
- [5] D. P. BERTSEKAS, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
- [6] C. L. BLOEBAUM AND P. HAJELA, *Heuristic Decomposition for Non-Hierarchic Systems*, AIAA/ASME/ASCE/AHS 32nd Structures, Structural Dynamics, and Materials Conference, Baltimore, MD, April, 1991, pp. 344–353.

- [7] G. DI PILLO, F. FACCHINEI AND L. GRIPPO, *A RQP algorithm using a differentiable exact penalty function for inequality constrained problems*, Technical Report 03.89, Dept. of Information and Systems, Rome, Italy, 1989.
- [8] R. FLETCHER, *Practical methods of optimization*, John Wiley and Sons, New York, 1980.
- [9] J. C. GILBERT, *Maintaining the positive definiteness of the matrices in reduced secant methods for equality constrained optimization*, IIASA working paper WP-87-123, IIASA, Laxenburg, Austria, 1987.
- [10] P. E. GILL, W. MURRAY, M. A. SAUNDERS AND W. H. WRIGHT, *Inertia-controlling methods for quadratic programming*, SIAM Review, 33 (1991), pp. 1-36.
- [11] P. E. GILL, W. MURRAY, M. A. SAUNDERS AND W. H. WRIGHT, *Procedures for optimization problems with a mixture of bounds and general constraints*, ACM Transactions on Mathematical Software, 10 (1984), pp. 282-298.
- [12] P. E. GILL, W. MURRAY AND M. A. SAUNDERS, *QPSOL*, Department of Operations Research, Stanford University, Stanford, CA, 1984.
- [13] G. KREISSELMEIER AND R. STEINHAUSER, *Systematic control design by optimizing a vector performance index*, Proc. IFAC Symp. on Computer Aided Design of Control Systems, Zurich, Switzerland, (1979), pp. 113-117.
- [14] D. C. LIU AND J. NOCEDAL, *On the limited memory BFGS method for large scale optimization*, Technical Report NAM 03, Dept. of EE and CS, Northwestern Univ., Evanston, IL, 1988.
- [15] B. A. MURTAGH AND M. A. SAUNDERS, *A projected Lagrangian algorithm and its implementation for sparse nonlinear constraints*, Reprint of Math. Prog. Study 16 (1982), pp. 84-117.
- [16] B. A. MURTAGH AND M. A. SAUNDERS, *Large scale linearly constrained optimization*, Reprint of Math. Prog. 14 (1978), pp. 41-72.
- [17] B. A. MURTAGH AND M. A. SAUNDERS, *MINOS 5.0 User's Guide*, Report SOL 83-20, Department of Operations Research, Stanford University, Stanford, CA, 1983.
- [18] E. R. PANIER AND A. L. TITS, *On feasibility, descent and superlinear convergence in inequality constrained optimization*, Technical Research Report 89-27, Systems Research Center, Univ. of Maryland, College Park, Maryland, 1989.
- [19] J. E. RENAUD AND G. A. GABRIELE, *Sequential global approximation in non-hierarchic system decomposition and optimization*, Advances in Design Automation, 1 (1991), pp. 191-200.
- [20] Y. S. SHIN, R. T. HAFTKA, L. T. WATSON, AND R. H. PLAUT, *Tracing structural optima as a function of available resources by a homotopy method*, Comput. Methods Appl. Mech. Engrg., 70 (1988), pp. 151-164.
- [21] Y. S. SHIN, R. T. HAFTKA, L. T. WATSON, AND R. H. PLAUT, *Design of laminated plates for maximum buckling load*, J. Composite Materials, 23 (1989), pp. 348-369.
- [22] J. SOBIESZCZANSKI-SOBIESKI, *Optimization by decomposition: a step from hierarchic to non-hierarchic systems*, Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Sept., 1988, Hampton, VA.
- [23] J. SOBIESZCZANSKI-SOBIESKI, *A linear decomposition method for large optimization problems*, NASA TM 83248, Feb., 1982.
- [24] J. SOBIESZCZANSKI-SOBIESKI, B. B. JAMES, AND M. F. RILEY, *Structural sizing by generalized, multilevel optimization*, AIAA J., 25(1) (1987), p. 139-145.

- [25] J. SOBIESZCZANSKI-SOBIESKI, *On the sensitivity of complex, internally coupled systems*, AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference, Williamsburg, VA, April, 1988.
- [26] J. SOBIESZCZANSKI-SOBIESKI, J.-M. F. BARTHELEMY, AND K. M. RILEY, *Sensitivity of optimum solutions to problem parameters*, AIAA J., 21 (1982), pp. 1291–1299.
- [27] G. N. VANDERPLAATS, Y. J. YANG, AND D. S. KIM, *An efficient multilevel optimization method for engineering design*, AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference, Williamsburg, VA, April, 1988, pp. 125–132.
- [28] G. VASUDEVAN, L. T. WATSON, AND F. H. LUTZE, *A homotopy approach for solving constrained optimization problems*, Tech. Rep. 88–50, Dept. of Computer Sci., VPI&SU, Blacksburg, VA, 1988.
- [29] L. T. WATSON, R. T. HAFTKA, F. H. LUTZE, R. H. PLAUT, AND P. Y. SHIN, *The application of globally convergent homotopy methods to nonlinear optimization*, Advances in Numerical Partial Differential Equations and Optimization, S. Gómez, J. P. Hennart, and R. A. Tapia (eds.), SIAM, Philadelphia, PA, 1991, pp. 284–298.
- [30] L. T. WATSON, S. C. BILLUPS, AND A. P. MORGAN, *HOMPACK: A suite of codes for globally convergent homotopy algorithms*, Tech. Rep. 85–34, Dept. of Industrial and Operations Eng., Univ. of Michigan, Ann Arbor, MI, 1985, and ACM Trans. Math. Software, 13 (1987), pp. 281–310.
- [31] L. T. WATSON AND W. H. YANG, *Optimal design by a homotopy method*, Applicable Anal., 10 (1980), pp. 275–284.
- [32] G. A. WRENN AND A. R. DOVI, *Multilevel decomposition approach to the preliminary sizing of a transport aircraft wing*, AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference, Williamsburg, VA, April, 1988.